

บทที่ 5

การออกแบบ และการทำงานของโปรแกรม

1. แนวคิดและหลักการ

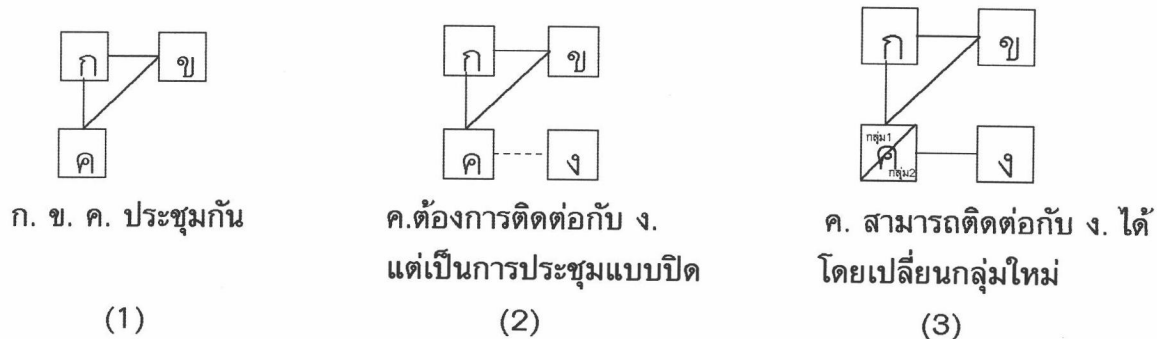
จากบทที่แล้วจะเห็นได้ว่าการติดต่อสื่อสารกันบนเน็ตแวย์นั้น สามารถทำได้ 2 วิธี คือ การใช้บริการรับ-ส่งข้อความ และการใช้บริการด้านติดต่อสื่อสาร แต่จะเห็นได้ว่า การใช้บริการรับ-ส่งข้อความ เหมาะกับการรับส่งข้อมูลที่เป็นข้อความยาวๆ ทีละข้อความ และมีประสิทธิภาพในการส่งต่ำจึงไม่เหมาะกับระบบนี้ ซึ่งจะต้องเป็นการติดต่อสื่อสารกันแบบโต้ตอบตัวอักษรต่อตัวอักษรและติดต่อกับผู้ใช้หลายคน จึงต้องการโปรโตคอลที่สามารถส่งข้อมูลได้ถูกต้อง และรวดเร็ว จึงได้เลือกใช้บริการด้านติดต่อสื่อสาร ซึ่งเป็นการส่งข้อมูลแบบลיתיเท่าเทียมกัน ซึ่งมีความรวดเร็ว และสามารถรองรับผู้ใช้ได้เป็นจำนวนมาก

ในส่วนของการใช้บริการด้านติดต่อสื่อสารเองก็มีโปรโตคอล ให้ใช้ได้คือ IPX และ SPX IPX เป็นโปรโตคอลที่ไม่รับรองการส่งและไม่ต้องสร้างเส้นทางเชื่อมต่อถึงกัน (Connectionless Oriented Protocol) ซึ่งเปรียบเสมือนการส่งจดหมาย ส่วน SPX เป็นโปรโตคอลที่มีลำดับการส่งและต้องสร้างเส้นทางเชื่อมต่อถึงกันก่อนเริ่มส่งข้อมูล (Connection Oriented Protocol) ซึ่งเปรียบเสมือนการโทรศัพท์ถึงกัน โดยการวิจัยนี้ได้เลือกใช้โปรโตคอล SPX เป็นโปรโตคอลหลักในการรับส่งข้อมูลระหว่างกัน และใช้โปรโตคอล IPX เป็นโปรโตคอลสำหรับการใช้ในการส่งข้อมูลเริ่มต้นก่อนที่จะสร้างเส้นทางเชื่อมต่อ (Connection) ถึงกัน เมื่อสร้างเส้นทางเชื่อมต่อแล้วจะให้โปรโตคอล SPX ในการรับส่งข้อมูล

เพื่อลดข้อยุ่งยาก และ ชับซ้อน ในการติดตั้งและใช้งาน จึงได้ออกแบบระบบโดยที่ไม่จำเป็นต้องมีการฝังโปรแกรมไว้ในหน่วยความจำ เพื่อคอยรอรับการเรียก แต่จะใช้วิธีการส่งข้อความแบบเดียวกับคำสั่ง SEND ของเน็ตแวย์ คือ เมื่อมีผู้เรียกเข้ามาก็จะมีข้อความปรากฏขึ้นที่บรรทัดล่างสุดของจอภาพ ถ้าต้องการตอบรับ ก็เพียงแต่กดปุ่ม Ctrl + Enter เพื่อลบข้อความ และเรียกโปรแกรมขึ้นมาใช้งานตามด้วยพารามิเตอร์ /r เพื่อเป็นการตอบรับ ก็จะสามารถติดต่อสื่อสารกันได้

เมื่อมีการติดต่อกันมากกว่า 2 คนขึ้นไป ก็เกิดปัญหาว่า ผู้ใช้คนที่ 3 ที่จะเข้ามาเป็นสมาชิกใหม่จะสามารถติดต่อกับสมาชิกอีกคนที่ติดต่อก่อนได้ด้วยหรือไม่ เช่น ก. ติดต่อกับ ข. อยู่ก่อนแล้วและ ก. ต้องการติดต่อกับ ค. อีก ข. และ ค. จะสามารถติดต่อกันได้ด้วยหรือไม่ แต่เนื่องจากวัตถุประสงค์ของการวิจัยนี้ ต้องการให้การติดต่อเป็นลักษณะของการประชุม ดังนั้นเมื่อมีผู้ใช้คนใหม่เข้ามาก็จะต้องสามารถ ติดต่อกับสมาชิกอื่น ๆ ในกลุ่มที่ติดต่อกันอยู่เดิมได้ด้วย โดยผู้ใช้คนใหม่ไม่จำเป็นต้องทำการเรียกเพื่อติดต่อกับสมาชิกในกลุ่มเอง การสร้างการติดต่อในส่วนนี้จะไปโดยอัตโนมัติ นั่นคือ จากตัวอย่างก่อนหน้านี้ เมื่อ ก. เรียก ค. ข. และ ค. จะมีการสร้างการติดต่อกันโดยอัตโนมัติ ดังรูปที่ 5.1

เช่น ง. ได้ แต่ ค. สามารถเปลี่ยนกลุ่มการประชุม เป็นกลุ่มที่ 2 และสามารถติดต่อกับ ง. ได้ โดยการติดต่อนี้จะไม่เกี่ยวข้องกับการประชุมของกลุ่มแรก ดังรูป



รูปที่ 5.3 แสดงการเปลี่ยนกลุ่มการติดต่อ

และเพื่อความสะดวกในการใช้งาน จึงได้ออกแบบให้มีการเรียกแบบกลุ่ม โดยกลุ่มในที่นี้ ได้อาศัยกลุ่มของผู้ใช้บนเน็ตเวิร์ก ซึ่งโดยปกติผู้ใช้งานเน็ตเวิร์กจะมีการแบ่งออกเป็นกลุ่มอยู่แล้ว เช่น กลุ่มผู้ใช้ฝ่ายบัญชี กลุ่มผู้ใช้ฝ่ายคอมพิวเตอร์ การเรียกเพียงแต่ระบุเงื่อนไขว่าเป็นการเรียกแบบกลุ่ม และระบุชื่อกลุ่ม โปรแกรมจะไปค้นหาเองโดยอัตโนมัติ ว่ามีใครบ้างที่อยู่ในกลุ่มนั้น และกำลังใช้งานในระบบเน็ตเวิร์กอยู่

2. การออกแบบโปรแกรม

เพื่อให้การพัฒนาโปรแกรมเป็นไปอย่างสะดวกและเป็นขั้นตอน จึงได้แบ่งโปรแกรมออกเป็นส่วนๆ โดยโปรแกรมจะมีส่วนประกอบหลักๆ 5 ส่วนคือ

- 1) ส่วนการเริ่มต้นการส่งโดยเป็นผู้เรียก
- 2) ส่วนการเริ่มต้นการส่งโดยเป็นผู้ตอบรับ
- 3) ส่วนการติดต่อ
- 4) ส่วนการแสดงผลทางจอภาพ
- 5) ส่วนการเลิกการติดต่อ

1) ส่วนการเริ่มต้นการติดต่อโดยเป็นผู้เรียก การทำงานขั้นแรกเมื่อรับชื่อผู้ใช้ที่ต้องการจะติดต่อดำเนินการแล้ว สถานีงานนั้นจะส่งคำร้องขอ (Request) ไปยังเครื่องบริการแฟ้มเพื่อหาตำแหน่งของสถานีงานที่ผู้ใช้คนที่เราต้องการจะติดต่อดำเนินการอยู่ เมื่อได้ตำแหน่งแล้วจะส่งข้อความด้วยการให้บริการเกี่ยวกับข้อความ (Message Service) ไปยังสถานีงานนั้นเพื่อบอกให้ทราบว่ามีผู้ต้องการจะติดต่อดำเนินการ แล้วจึงส่งข้อมูลตามโครงสร้างการส่งข้อมูลให้ผู้ถูกเรียก (ดูข้อ 3.3) ด้วยโปรโตคอล IPX เพื่อบอกตำแหน่งสถานีงานของตัวเองให้ผู้ถูกเรียกทราบ และในกรณีที่สมาชิกในกลุ่มก็จะส่งตำแหน่งของสถานีงานของสมาชิกในกลุ่มไปด้วย และจะทำการส่งข้อมูลไปบอกสมาชิกในกลุ่ม (ดู ข้อ 3.4 การส่งข้อมูลระหว่างติดต่อ) ให้เตรียมตัวรับการติดต่อจากสมาชิกใหม่ หลังจากนั้นจะคอยการตอบรับจากผู้ถูกเรียก เมื่อผู้ถูกเรียกตอบกลับมาก็จะทำการสร้างเส้นทางการเชื่อมต่อ (Connection) เก็บค่าต่างลงในตารางการติดต่อ (ตารางที่ 5.1) และเตรียมตัวเข้าสู่การ



ทำงานในส่วนการติดต่อ ตามข้อ 3)

2) ส่วนการเริ่มต้นการติดต่อโดยเป็นผู้ตอบรับ เนื่องจากการออกแบบให้ไม่ต้องมีส่วนของโปรแกรมฝั่งตัวในหน่วยความจำ ข้อความที่ส่งมาจากผู้เรียกด้วย Message service จะถูกแสดงที่บรรทัดล่างของจอภาพ โดยเซลล์ของเน็ตเวิร์กเหมือนกับการใช้คำสั่ง Send ดังนั้นในการตอบรับ ผู้ตอบรับต้องระบุเงื่อนไขว่าต้องการตอบรับการเรียก โปรแกรมส่วนนี้จึงถูกเรียกขึ้นมาทำงาน และจะทำหน้าที่คอยรับข้อมูลจากผู้เรียกส่งมาด้วยโปรโตคอล IPX (ดูข้อ 3.3) เมื่อได้รับข้อมูลแล้ว ในข้อมูลนั้นจะบอกได้ว่าใครเป็นผู้เรียก และมีใครเป็นสมาชิกในกลุ่มบ้าง หลังจากนั้นก็จะตอบกลับเพื่อให้ผู้เรียกรู้ และทำการสร้างเส้นทางการเชื่อมต่อไปกับผู้เรียก และถ้ามีสมาชิกในกลุ่มก็จะทำการสร้างการติดต่อกับสมาชิกในกลุ่มทีละคน และเก็บข้อมูลต่างลงในตารางการติดต่อแล้วเตรียมตัวเข้าสู่การทำงานในส่วนการติดต่อตามข้อ 3)

3) ส่วนการติดต่อ หลังจากที่มีการสร้างเส้นทางการเชื่อมต่อจากข้อ 1 และ 2 แล้วส่วนนี้จะมีหน้าที่หลัก 2 อย่างคือ

- ส่วนการรอรับการกดแป้นพิมพ์
- ส่วนการตรวจสอบข้อมูลที่ส่งมา

ส่วนการรอรับการกดแป้นพิมพ์ ส่วนนี้จะทำหน้าที่คอยรับตัวอักษรจากแป้นพิมพ์ แล้วส่งออกไปให้กับสมาชิกในกลุ่มทันที นั่นคือเมื่อกด 1 ตัวอักษร ผู้รับก็จะได้รับตัวอักษรนั้นทันที ซึ่งการส่งตัวอักษรนี้ จะส่งค่าของ Destination Index ในตารางการติดต่อ (ตารางที่ 5.1) ไปด้วย ค่านี้จะบอกตำแหน่งของเรคคอร์ดในตารางการติดต่อกับคู่การติดต่อนั้น ซึ่งจะทำให้ผู้รับสามารถนำตัวอักษรที่ได้รับไปแสดงผลได้ถูกหน้าต่าง

ส่วนการตรวจสอบข้อมูลที่ส่งมา จะคอยตรวจสอบในแถวคอยแบบวงกลม (รูปที่ 5.4) ของการรับข้อมูลว่ามีข้อมูลหรือไม่ เนื่องจากข้อมูลที่รับเข้ามา จะถูกนำไปเก็บไว้ในแถวคอยนี้ก่อน ถ้ามีข้อมูลในแถวคอยก็จะส่งข้อมูลต่อไปยัง ส่วนการแสดงผลทางจอภาพ เพื่อแสดงข้อมูลที่มีผู้ส่งมาให้ โดยในการรับและส่งข้อมูลนี้ จะอาศัยตารางการติดต่อในการหาว่าสมาชิกในกลุ่มมีใครบ้างเพื่อที่จะส่งข้อมูลไปให้ และเพื่อที่จะรู้ว่าข้อมูลที่รับเข้ามาเป็นของใคร

4) ส่วนการแสดงผลทางจอภาพ เนื่องจากโปรแกรมนี้อาจสามารถติดต่อกับผู้ใช้ได้หลายคนพร้อมกัน ดังนั้นจึงต้องมีการแบ่งจอภาพออกเป็นส่วนๆ สำหรับผู้ใช้แต่ละคน และอีกส่วนสำหรับข้อมูลของตัวเองที่ป้อนเข้าไป โดยที่แต่ละหน้าต่างจะมีบัฟเฟอร์สำหรับเก็บข้อความที่ถูกแสดงที่หน้าต่างนั้น (ตารางที่ 5.2) โดยสามารถเก็บข้อความไว้ได้ 5 บรรทัด บรรทัดละ 85 ตัวอักษร เนื่องจากข้อมูลที่ส่งมาจะมีค่า Destination index ส่งมาด้วย จึงทำให้ทราบตำแหน่งของหน้าต่างที่จะแสดงผล โดยดูจากโครงสร้างข้อมูลของหน้าต่างการแสดงผล ซึ่งตำแหน่งของเรคคอร์ดของโครงสร้างการแสดงผลจะตรงกับตำแหน่งของเรคคอร์ดของตารางการติดต่อ

5) ส่วนการเลิกการติดต่อ เมื่อกดปุ่มเลิกการติดต่อ โปรแกรมจะทำการส่งข้อความไปบอกสมาชิกในกลุ่มว่าต้องการเลิกการติดต่อ จากนั้นจะทำการสิ้นสุดการติดต่อ (Terminate Connection) แล้วปล่อยหน่วยความจำส่วนที่นำมาใช้ในการสร้างการติดต่อ ลบค่าในตารางการติดต่อและโครงสร้างข้อมูลของหน้าต่าง แล้วแสดงผลหน้าจอใหม่ โดยการเลิกการติดต่อต้องเลิกเป็นกลุ่ม ไม่สามารถเลิกกับผู้ใช้เป็นรายๆได้



3. โครงสร้างข้อมูล

3.1 ตารางการติดต่อ ตารางการติดที่จะเก็บข้อมูลที่เกี่ยวข้องกับการติดต่อของสมาชิกทั้งหมดในกลุ่ม เช่น ชื่อผู้ใช้ ตำแหน่งของสถานีงาน เป็นต้น โดยที่ตารางนี้จะมีจำนวนเรคคอร์ดได้สูงสุด 20 เรคคอร์ด นั่นคือจะสามารถติดต่อกับผู้ใช้อื่นๆ ได้พร้อมกัน 20 คน โครงสร้างของตารางการติดต่อดังนี้

ค่าที่เก็บ	ชนิด	ความหมาย
connectionID	WORD	เก็บ Connection ID ที่ต่อกับเครื่องบริการเพิ่ม
group	BYTE	เก็บหมายเลขกลุ่มในการติดต่อ (0 หมายถึงว่าง)
destinationIndex	WORD	เก็บตำแหน่งของเรคคอร์ดของสถานีงานตัวเอง
connection	WORD	เก็บหมายเลข Connection ของการติดต่อด้วย SPX
network	BYTE[4]	เก็บหมายเลข Network
node	BYTE[6]	เก็บหมายเลข Node
socket	WORD	เก็บหมายเลข Socket
ECBAddress	pointer	ชี้ไปยังโครงสร้างของ ECB
SPXAddress	pointer	ชี้ไปยังโครงสร้างของ SPX
bufferAddress	pointer	ชี้ไปยัง Buffer ที่เก็บข้อมูลที่ส่งมา
username	char[25]	เก็บชื่อผู้ใช้

ตารางที่ 5.1 แสดงโครงสร้างของตารางการติดต่อ

3.2 ตารางการแสดงผล จะเก็บข้อมูลเกี่ยวกับการแสดงผล โดย 1 หน้าต่างจะมีข้อมูล 1 เรคคอร์ด ตำแหน่งของเรคคอร์ดในตารางการแสดงผล จะตรงกับตำแหน่งของเรคคอร์ดในตารางการติดต่อ นั่นคือข้อมูลของสมาชิกที่อยู่ในตารางการติดต่อเรคคอร์ดที่ 1 จะถูกแสดงผลตามข้อมูลของเรคคอร์ดที่ 1 ในตารางการแสดงผล และขนาดของหน้าต่างจะปรับโดยอัตโนมัติเมื่อมีการเพิ่มหรือลดจำนวนสมาชิกในกลุ่ม

ค่าที่เก็บ	ชนิด	ความหมาย
left	int	บอกตำแหน่งซ้ายสุดของหน้าต่าง
top	int	บอกตำแหน่งบนสุดของหน้าต่าง
right	int	บอกตำแหน่งขวาสุดของหน้าต่าง
bottom	int	บอกตำแหน่งล่างสุดของหน้าต่าง
x,y	int	เก็บตำแหน่งเคอร์เซอร์ของหน้าต่างนั้น
ptrx	int	เก็บตำแหน่งที่จะเก็บข้อมูลในบัฟเฟอร์
buffer[5][85]	char	เก็บข้อมูลของหน้าต่างนั้น

ตารางที่ 5.2 ตารางการแสดงผล

3.3 โครงสร้างการส่งข้อมูลให้ผู้ถูกเรียกด้วย IPX ใช้ในการส่งข้อมูลของผู้เรียก และสมาชิกในกลุ่มไปให้ผู้รับ มีโครงสร้างดังนี้

```
[n][my IPX address][my connection][my username][connection type]
[network][node][username]      )
[network][node][username]      |
•                               }   n
•                               |
•                               )
```

n หมายถึง จำนวน สมาชิกในกลุ่ม

my IPX address หมายถึง หมายเลขเครือข่าย ตำแหน่งสถานีงาน และหมายเลข
ชื่อคีย์ของผู้เรียก

my connection หมายถึง หมายเลขการเชื่อมต่อกับเครื่องบริการแฟ้มของผู้เรียก

my username หมายถึง ชื่อผู้ใช้ของผู้เรียก

connection type หมายถึง ชนิดการติดต่อ 0 คือ การติดต่อแบบเปิด 1 คือการติดต่อ

แบบปิด

network หมายถึง หมายเลขเครือข่ายของสมาชิก

node หมายถึง ตำแหน่งของสถานีงานของสมาชิก

username หมายถึง ชื่อผู้ใช้ของสมาชิก

3.4 การส่งข้อมูลระหว่างการติดต่อ เป็นการส่งข้อมูลระหว่างสมาชิกในกลุ่มเมื่อมีการ
สร้างการติดต่อถึงกันแล้ว และใช้ส่งข้อมูลเพื่อบอกการควบคุมการติดต่อในกลุ่ม เช่น การสิ้นสุด
การติดต่อ การรับสมาชิกใหม่ เป็นต้น ข้อมูลประกอบด้วย

Index	Character
-------	-----------

Index หมายถึง ตำแหน่งของเรคคอร์ดในตารางการเชื่อมต่อของผู้รับ

Character หมายถึง ตัวอักษรที่ส่งมา เช่น A,B,C,ก,ข,ค เป็นต้น

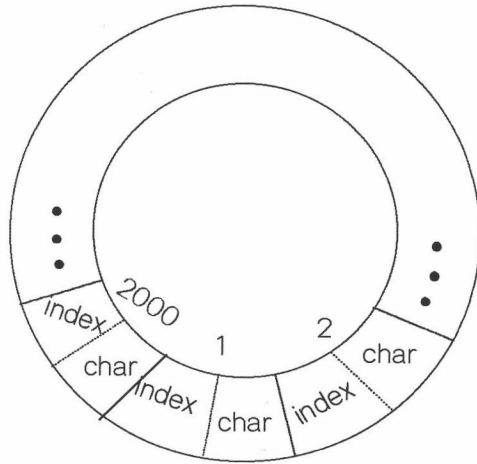
หรือ ค่าควบคุม คือ มีค่าเป็น 1 หมายถึง ให้รอรับสมาชิกใหม่

2 หมายถึง เปลี่ยนกลุ่มการสนทนาเป็นแบบปิด

3 หมายถึง เปลี่ยนกลุ่มการสนทนาเป็นแบบเปิด

27 หมายถึง ยกเลิกการติดต่อ

3.5 การเก็บข้อมูลที่สมาชิกอื่นส่งมา ข้อมูลที่ส่งมาจากสมาชิกในกลุ่ม จะถูกนำมาเก็บ
ไว้ในแถวคอยแบบวงกลมก่อน และโปรแกรมในส่วนการติดต่อจะไปดึงข้อมูลมาจากแถวคอย



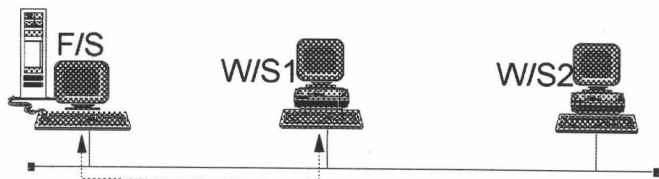
รูปที่ 5.4 แถวคอยแบบวงกลมของการรับข้อมูล

4. การทำงานของโปรแกรม

การทำงานในส่วนหลักของโปรแกรม จะเป็นส่วนของการรับส่งข้อความกันโดยใช้ API ใน ส่วนของ IPX/SPX ซึ่งโปรแกรมนี้อาจใช้โปรโตคอล IPX สำหรับรับส่งข้อมูลบางอย่าง เช่นตำแหน่ง (Address) ของคู่สนทนา ก่อนที่จะมีการสร้างเส้นทางสำหรับการติดต่อสื่อสารสำหรับโปรโตคอล SPX ขึ้น เมื่อสร้างเส้นทางสำหรับการติดต่อสื่อสารขึ้นแล้ว การรับส่งข้อความจะใช้โปรโตคอล SPX ตลอด ซึ่งลักษณะการทำงานโดยย่อเป็นดังนี้



ขั้นตอนการทำงานของโปรแกรมเมื่อผู้ใช้ที่สถานีงานที่ 1 เรียกโปรแกรมขึ้นมาทำงาน และต้องการติดต่อกับผู้ใช้ซึ่งอยู่ที่สถานีงานที่ 2



ขั้นตอนที่ (1)

สถานีงานที่ 1 จะใช้ API ในส่วนของ Connection Service ติดต่อกับ File Server เพื่อหาตำแหน่งของสถานีงานที่ผู้ใช้คนที่เราต้องการจะติดต่อด้วยใช้งานอยู่ ในที่นี้สมมติให้อยู่ที่สถานีงานที่ 2



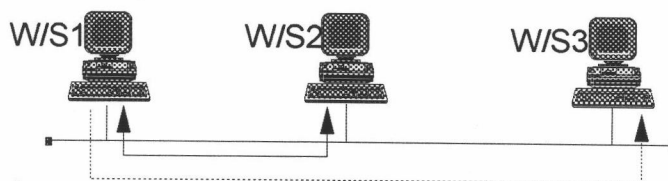
เมื่อได้ตำแหน่งของสถานีงาน จากขั้นตอนที่ (1) โปรแกรมก็จะส่งข้อความไปที่สถานีงานที่ 2 เพื่อบอกให้รู้ว่า มีผู้ต้องการติดต่อด้วย ข้อความที่ส่งไปตอนนี้จะไปแสดงที่บรรทัดล่างสุดของจอภาพของสถานีงานที่ 2 เหมือนการใช้คำสั่ง SEND ของ NetWare หลังจากนั้นโปรแกรมจะส่งข้อมูลโดยใช้โปรโตคอล IPX เพื่อบอกตำแหน่งสถานีงาน และ Username ของตัวเองให้สถานีงานที่ 2 รู้ โดยจะส่งข้อมูลออกไปเป็นระยะๆ



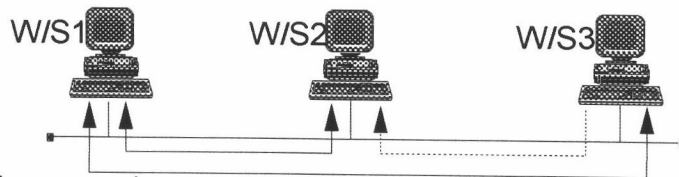
เมื่อสถานีงานที่ 2 ได้รับข้อความว่า มีผู้ต้องการติดต่อด้วยที่บรรทัดล่างสุดของจอภาพและเรียกโปรแกรมขึ้นมาทำงานเพื่อตอบรับการเรียก สถานีงานที่ 2 ก็จะได้รับข้อมูลที่ส่งมาโดยโปรโตคอล IPX ก็จะทำให้ทราบตำแหน่งสถานีงานของผู้ที่เรียกมา หลังจากนั้นสถานีงานที่ 2 จะทำการสร้างเส้นทางสำหรับการติดต่อสื่อสาร (Connection) ขึ้น และจะมีการให้หมายเลขของเส้นทาง (Connection ID) นั้นกับสถานีงานทั้ง 2 ฝ่ายเพื่อใช้อ้างอิงในการส่งข้อความถึงกันต่อไป

รูปที่ 5.5 แสดงการทำงานของโปรแกรมเมื่อมีการติดต่อกัน

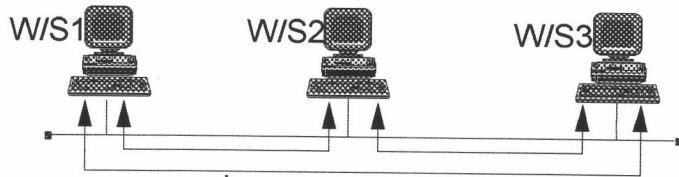
การทำงานเมื่อมีการเรียกแบบอัตโนมัติ คือเมื่อมีสมาชิกเก่าอยู่แล้ว และต้องการติดต่อกับสมาชิกใหม่ โดยสมาชิกใหม่จะสามารถติดต่อกับสมาชิกเก่าคนอื่นๆ ได้โดยอัตโนมัติจะเพิ่มขั้นตอนมากกว่าปกติ คือ การสร้างการติดต่อระหว่างสมาชิกใหม่กับสมาชิกเก่าอื่น ๆ มีขั้นตอนดังนี้



สถานีงานที่ 1 และ 2 กำลังติดต่อกัน และสถานีงานที่ 1 ต้องการติดต่อกับสถานีงานที่ 3 ขั้นตอนการติดต่อส่วนแรกนี้จะเหมือนกับการติดต่อตามขั้นตอนที่กล่าวมาแล้ว แต่ข้อมูลที่สถานีงานที่ 1 ส่งไป กับโปรโตคอล IPX จะมีข้อมูลส่วนหนึ่งบอกว่ามีสมาชิกอื่นอยู่ด้วย และในขณะเดียวกันก็จะส่งข้อมูลไปบอกสมาชิกในกลุ่มว่า ให้เตรียมรับการติดต่อเนื่องจากมีสมาชิกใหม่เข้ามา



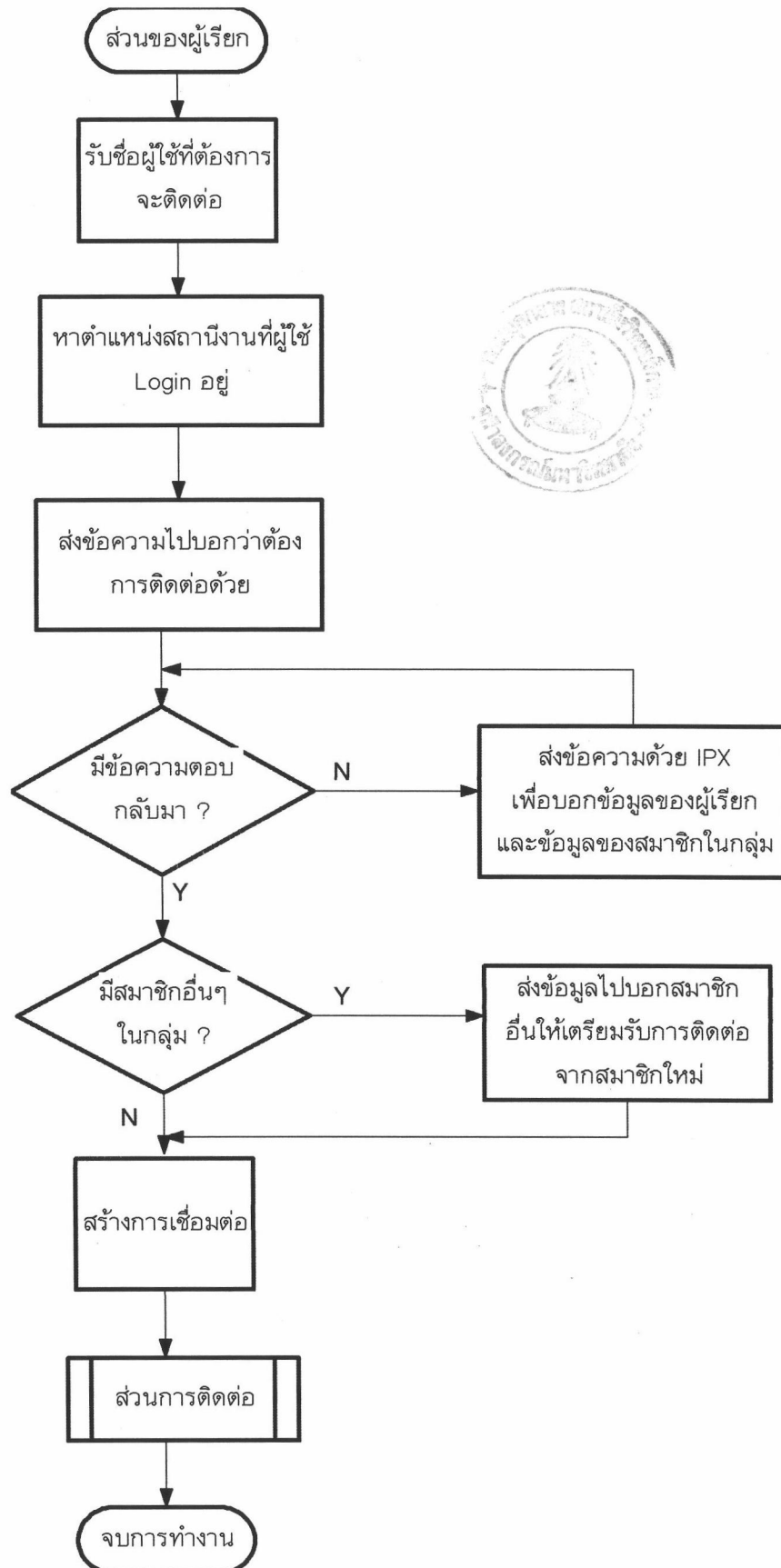
หลังจากที่สถานีงานที่ 1 ติดต่อกับสถานีงานที่ 3 ได้แล้ว สถานีงานที่ 3 จะเริ่มทำการติดต่อกับสถานีงานที่ 2 ที่รออยู่ โดยอาศัยข้อมูลจากโปรโตคอล IPX ที่ได้รับจากสถานีงานที่ 1



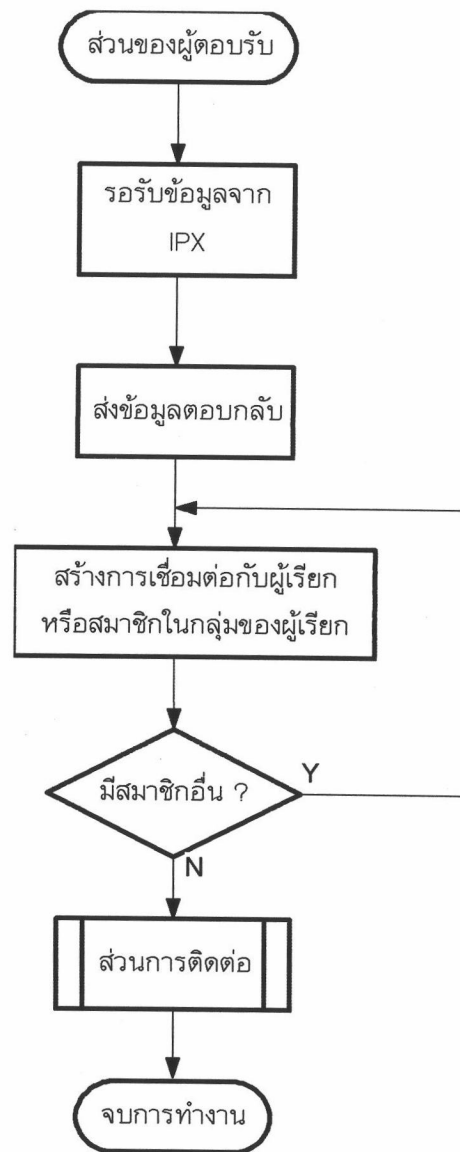
สถานีทั้ง 3 สามารถติดต่อสื่อสารกันได้

รูปที่ 5.6 แสดงการทำงานของโปรแกรมเมื่อมีการติดต่อกันแบบอัตโนมัติ

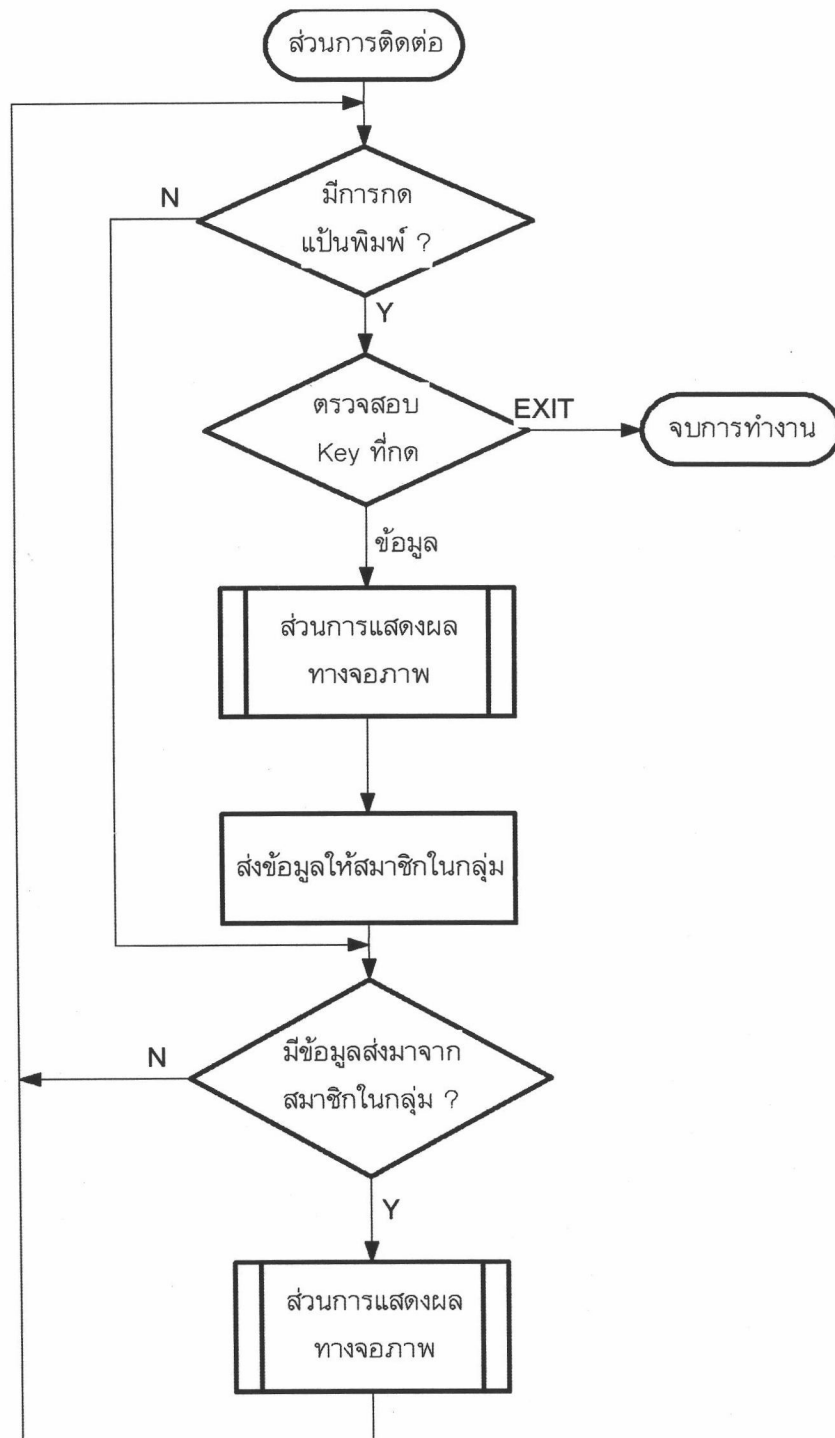
การทำงานของโปรแกรมในส่วนหลัก 5 ส่วน สามารถเขียนเป็นแผนผังแสดงขั้นตอนการทำงานได้ดังรูปที่ 5.6-5.9



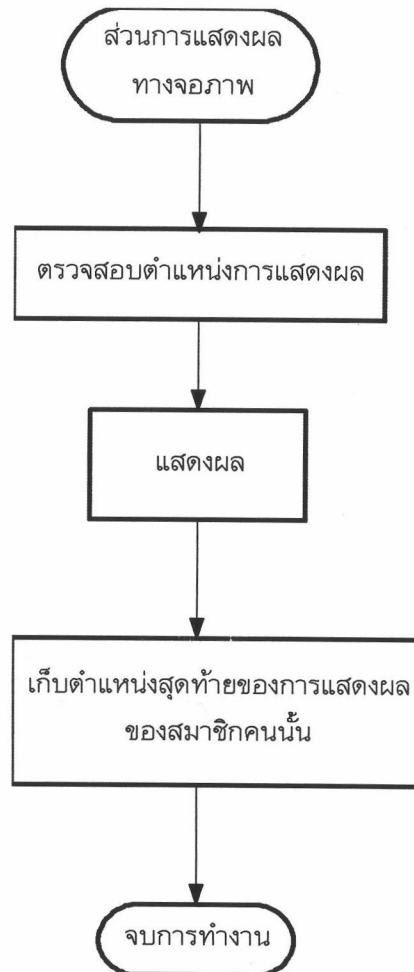
รูปที่ 5.7 แสดงขั้นตอนการทำงานของโปรแกรมส่วนของผู้เรียก



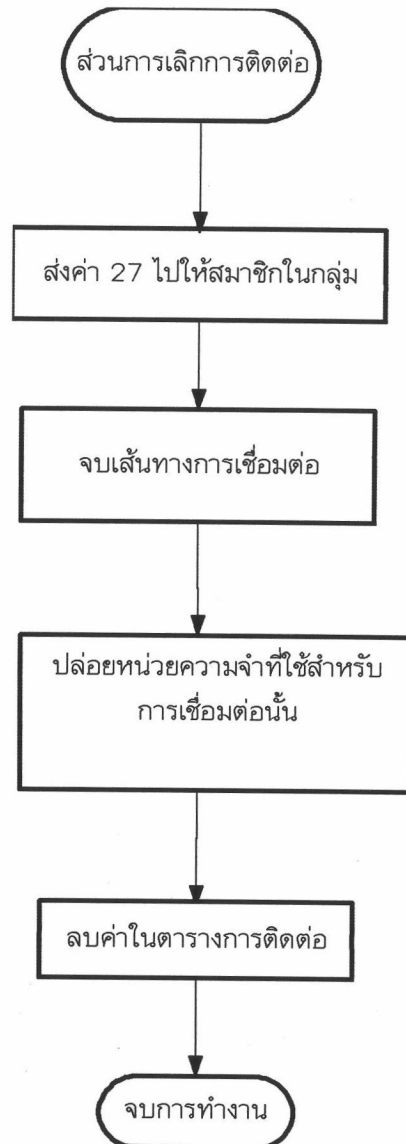
รูปที่ 5.8 แสดงขั้นตอนการทำงานของโปรแกรมส่วนของผู้ตอบรับ



รูปที่ 5.9 แสดงขั้นตอนการทำงานของโปรแกรมส่วนการติดต่อ



รูปที่ 5.10 แสดงขั้นตอนการทำงานของโปรแกรมส่วนการแสดงผล



รูปที่ 5.11 แสดงขั้นตอนการทำงานของโปรแกรมส่วนการเลือกการติดต่อ